



Docket No. 50277-0236 (OID 1998-02-01)

PATENT

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Application of	:	
	:	
Alok Kumar SRIVASTAVA et al.	:	Confirmation Number: 3268
	:	
Serial No.: 09/258,013	:	Group Art Unit: 2142
	:	
Filed: February 25, 1999	:	Examiner: Beatriz Prieto
	:	
For: DETERMINING THE	:	
PARTICIPANTS IN A DISTRIBUTED	:	
OPERATION	:	

---

Mail Stop Appeal Brief-Patents  
Commissioner of Patents  
P. O. Box 1450  
Alexandria, VA 22313-1450

**APPEAL BRIEF**

Sir:

This Appeal Brief is submitted in support of the Notice of Appeal filed June 29,  
2004.

**I. REAL PARTY IN INTEREST**

Oracle International Corporation is the real party in interest.

## **II. RELATED APPEALS AND INTERFERENCES**

Appellants are unaware of any related appeals and interferences.

## **III. STATUS OF CLAIMS**

Claims 1-9, 11-19, and 22-31 are pending in the application.

Claims 1-9, 11-19, and 22-31 have been finally rejected in the Final Office Action mailed May 3, 2004.

Specifically, Claims 1, 11, 22, 24, 27, and 29 have been rejected under 35 U.S.C. §112, first paragraph, as allegedly failing to comply with the written description requirement.

Claims 1-9, 11-19, 22-23, 25-28, and 30-31 have been rejected under 35 U.S.C. §103(a) as allegedly unpatentable over U.S. Patent Number 5,835,766 by Iba et al. ("*IBA*") in view of U.S. Patent Number 5,778,179 by Kanai et. al. ("*KANAI*").

Claims 1 and 11 have been rejected under 35 U.S.C. §103(a) as allegedly unpatentable over U.S. Patent Number 5,459,871 by Van Den Berg ("*VAN DEN BERG*") in view of U.S. Patent Number 4,412,285 by Neches et. al. ("*NECHES*").

Claim 1 has been rejected under 35 U.S.C. §102(b) as allegedly anticipated by *NECHES*.

Claims 1 and 11 have been rejected under 35 U.S.C. §101 as claiming the same invention as that of claims 1-3 of prior U.S. Patent No. 6,594,702 (Fischer et. al.). This is a statutory double-patenting rejection.

Claim 1 has been rejected under the judicially created doctrine of “obviousness-type” double patenting over claims 1-3 of *FISCHER*, et. al., U.S. Patent No. 6,594,702 in view of *KANAI*.

#### **IV. STATUS OF AMENDMENTS**

Claims 24 and 29 have been amended to comply with the objection as to form raised by the Examiner in the Final Office Action.

#### **V. EXPLANATION OF THE INVENTION**

One of the long-standing challenges in computing is the detection of deadlocks. A deadlock occurs if a set of entities exists such that each entity in the set is waiting for the release of at least one resource owned by another entity in the set. In the context of a database system, for example, such entities include processes and transactions. (Application, page 1, lines 8-13). In a distributed computer system, the resources and entities involved in deadlocks may be distributed among many nodes. Thus, in the example of a distributed database system, transaction T1 may reside on one node, while transaction T2 resides on a different node. (Application, page 2, lines 20-22). Furthermore, in a distributed database system there may exist a number of distributed operations, such as a distributed transaction which is a transaction executed by database servers that may reside on multiple nodes. (Application, page 3, lines 9-12; see also Application, page 5, lines 5-7).

Communication between database servers, especially those residing on different nodes, can involve a relatively large amount of overhead, and may substantially delay receipt by a deadlock detection handler of the information required to build a traditional

wait-for-graph (WFG) for deadlock detection. Often, the cost in overhead and delays is so great that deadlock handlers are configured to forego the (WFG) cycle technique when attempting to detect deadlocks that may involve distributed resources. (Application, page 3, lines 19-24).

To address the problems presented by the WFG cycle technique and by other prior approaches (e.g. the time-out technique), the present invention describes a mechanism and system for making available information that identifies participants of a distributed operation by registering the information with a name service. Once the participant information has been registered with the name service, the name service supplies the information to entities that request it. An example of a distributed operation is a distributed transaction executed by two or more database servers. However, it should be noted that there are many types of distributed operations, and the present invention is not limited to any particular type of distributed operation. (Application, page 5, lines 2-7; see also Application, page 7, lines 11-20).

The set of entities participating in a distributed operation is referred to as the “participant set”, and does not necessarily include all the entities participating in a distributed operation. Entities which may be members of a participant set include, for example, processes, transactions, database servers, and nodes on which processes reside. (Application, page 7, lines 21-25, emphasis added).

In one example, to register information with the name service and make information available to other name service clients, a participant in a distributed operation (a name service client) transmits a publication request to the name service. A publication request is a request to make information available to a set of name service clients that

request the information. Typically, the publication request includes a key and data associated with the key. The data associated with the key is referred to as published data because once the name service receives the published data and the associated key, the name service supplies the data to any name service client requesting data associated with the key. (Application, page 9, lines 5-12).

In another example, a publication request may be made by a coordinator process for a distributed transaction, where the publication request may specify the spanning set of entities of the distributed transaction and a operation key that may comprise the transaction id X. (Application, page 9, line 23 to page 10, line 4). The spanning set may change as a distributed transaction progresses. So that the published spanning set accurately reflects the actual members of the spanning set, the coordinator process may update the published spanning set at various stages throughout the life of the respected distributed transaction. (Application, page 10, lines 11-14).

While the present invention has been illustrated using a coordinator process that transmits publication requests to register participant set information, other types of entities may register participant set information. These other types of entities include slaves participating in a distributed transaction, and entities not participating in the distributed operation. (Application, page 10, line 25 to page 11, line 2).

The pending claims describe techniques for determining participants of a distributed transaction in a distributed system. For example, Claim 1 describes a method of determining participants of a distributed transaction, the method comprising the step of registering, in a name service, participant data that identifies a plurality of participants that are participating in said distributed transaction, wherein said step of registering

occurs in response to said plurality of participants commencing participation in said distributed transaction, and the step of causing a node that requires information about participants in said distributed transaction to request said participant data from said name service.

In another example, Claim 22 describes a method for determining a plurality of participants that are participating in a distributed transaction, the method comprising the computer-implemented step of receiving first data that identifies said plurality of participants in response to said plurality of participants commencing participation in said distributed transaction; the step of registering the first data in response to receiving it; the step of receiving a request from a node; and the step of providing, in response to the request from the node, a second data wherein the second data includes at least part of the first data.

## **VI. GROUNDS FOR REJECTION TO BE REVIEWED ON APPEAL**

Whether Claims 1, 11, 22, 24, 27, and 29 comply with the written description requirement under 35 U.S.C. §112, first paragraph.

Whether Claims 1, 3-6, 11, 13-16, and 22 are patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI*.

Whether Claims 1 and 11 are patentable under 35 U.S.C. §103(a) as being nonobvious over *VAN DEN BERG* in view of *NECHES*.

Whether the Office Actions have established the requisite suggestion, teaching, or motivation to combine the prior art references cited in the above obviousness rejections.

Whether Claim 1 is patentable under 35 U.S.C. §102(b) as not being anticipated by *NECHES*.

Whether Claims 1 and 11 are patentable over a statutory double-patenting rejection under 35 U.S.C. §101 with respect to Fischer et. al., U.S. Patent No. 6,594,702 ("*FISCHER*").

Whether Claim 1 is patentable over an obviousness-type double patenting rejection over claims 1-3 of *FISCHER* in view of *KANAI*.

## **VII. ARGUMENT**

### **A. Claims 1, 11, 22, 24, 27, and 29 comply with the written description requirement under 35 U.S.C. §112, first paragraph because the Application as filed expressly supports the challenged claim limitations**

To satisfy the written description requirement, a patent specification must describe the claimed invention in sufficient detail so that one skilled in the art can reasonably conclude that the inventor had possession of the claimed invention. *See Moba, B.V. v. Diamond Automation, Inc.*, 325 F.3d 1306, 1319, 66 USPQ2d 1429, 1438 (Fed. Cir. 2003); *Vas-Cath, Inc. v. Mahurkar*, 935 F.2d 1555, 1563, 19 USPQ2d 1111, 1116 (Fed. Cir. 1991); MPEP §2163. The fundamental factual inquiry is whether the specification conveys with reasonable clarity to those skilled in the art that, as of the filing date sought, applicant was in possession of the invention as now claimed. *See Vas-Cath*, 935 F.2d at 1563-64, 19 USPQ2d at 1117. Furthermore, with respect to new or amended claims, MPEP §2163, subsection I-B, expressly states that “[w]hile there is no *in haec verba*

requirement, newly added claim limitations must be supported in the specification through express, implicit, or inherent disclosure.”

The Final Office Action rejected claims 1, 11, 22, and 27 alleging that the added claim limitation “**wherein said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction**” was not supported by the applicant’s disclosure. To support the rejection, the Examiner states that since the described coordinator process, which registers participant set to name service, is not aware as to when a participant process “commenced participating” in a distributed transaction, it is upon transmitting a publication request and receiving response to this request, that this coordinator is only able to determine a process that is participating (not commencing or ending) in a distributed transaction.

The applicant respectfully submits that explicit support for the above limitation can be found in page 10, lines 14-21 of the Application:

For example, when a distributed transaction is **initiated**, the **coordinator process causes the spawning of one or more slave processes**, such as slaves 232, 234, and 236. Because all of the processes participating in the distributed transaction X are on database servers 210, 230, and 270, **coordinator process 222 transmits to name service 202 a publication request** that specifies (1) that the published spanning set is database servers 210, 230, and 270, and (2) that distributed transaction identifier X is the distributed operation key to associate with the published spanning set. (Emphasis added).

Thus, the above passage clearly and unambiguously specifies that registering of participants (in this example slave processes), by the coordinator process, occurs in response to said participants being spawned by the coordinator process to participate in the distributed transaction.



Furthermore, in page 9, line 23 to page 10, line 6, and with respect to Fig. 2 of the Application, it is stated:

In order to provide data indicating the members of the spanning set of a distributed transaction to a node requiring that information, data indicating members of the spanning set are registered with name service 202. **To register the data, coordinator process 222 transmits a publication request to name service 202**, such as publication request 224. Publication request 224 specifies a published spanning set and a distributed operation key. ... A distributed operation key is a key to associate with published data identifying participants of a distributed operation, such as a spanning set. The distributed operation key used in this example is the distributed transaction id X. A distributed transaction id is data that identifies a distributed transaction, such as distributed transaction X. (Emphasis added).

Thus, the passage cited above clearly manifests that, contrary to the Examiner's assertion that a coordinator process cannot register participants but can only find participants in a distributed transaction by querying the name service, the coordinator process can and does register participants in a distributed transaction with the name service. Furthermore, the passage cited above clearly indicates that the coordinator process can register the participants when the participants' participation in the distributed transaction is initiated since upon commencement of participation the coordinator is able to associate a participant with the distributed transaction ID.

For these reasons, the applicant respectfully submits that Claims 1, 11, 22, and 27 are supported by the disclosure as filed and do not violate the written description requirement set forth in 35 U.S.C. §112, first paragraph.

The Final Office Action also rejected claims 24 and 29 stating that the added claim limitation **"wherein said request is received after a particular participant of said plurality of participants has waited for a threshold period of time"** was not found to be supported by the applicant's disclosure. Specifically, the Examiner states that

“[i]t is not clear where in the specification is it particularly pointed out that a process that is participating in a distributed transaction waits for period of time”. (Final Office Action, page 3, paragraph 3).

In page 10, line 25 to page 11, line 2, the Application as filed states that:

While the present invention has been illustrated using a coordinator process that transmits publication requests to register participant set information, other types of entities may register participant set information. **These other types of entities include slaves participating in a distributed transaction, . . . .** (Emphasis added).

Further, in page 12, lines 7-14 with respect to Fig. 2, the Application states:

Deadlock handler 274 eventually determines that lock 276 indicates that slave 236 has been waiting for resource 275 a threshold period of time. After detecting that **slave 236 has been waiting a threshold period of time**, deadlock handler 274 begins the process of detecting whether slave 236 may be deadlocked using the cycle technique. Before generating the distributed wait-for-graph needed for the cycle technique, deadlock handler 274 determines the spanning set. **To determine the spanning set, deadlock handler 274 queries name service 202** for the published spanning set using distributed transaction id X. (emphasis added).

Thus, the two paragraphs cited above clearly convey to one of ordinary skill in the art that: (1) a slave process may be a participant in a distributed transaction, and (2) that a name service client (a deadlock handler) requests information from the name service after a particular participant (the slave process) has waited for a threshold period of time. For this reason, the applicant respectfully submits that Claims 24 and 29 are supported by the disclosure as filed and are in compliance with the written description requirement set forth in 35 U.S.C. §112, first paragraph.

Therefore, the rejections of Claims 1, 11, 22, 24, 27, and 29 under 35 U.S.C. §112, first paragraph lack the requisite factual and legal basis. Appellant respectfully submits that the imposed rejections under 35 U.S.C. §112, first paragraph are

**improper** and respectfully solicits the Honorable Board to **reverse** each of these imposed rejections.

**B. Claims 1, 3-6, 11, 13-16, and 22 are patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* because the combination of *IBA* and *KANAI* does not disclose, teach or suggest all of the claimed limitations**

This argument will be developed in the subheadings that follow.

**1. Independent Claim 1 is patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI***

As stated in MPEP § 2143.03: “To establish prima facie obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art.” In *Re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974).

Currently, Claim 1 reads as follows:

A method of determining participants of a distributed transaction in a distributed system, the method comprising the steps of:  
registering, in a name service, participant data that identifies a plurality of participants that are participating in said distributed transaction, wherein **said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction;**  
causing a node that requires information about participants in said **distributed transaction** to request said participant data from said name service.” (Emphasis added).

The applicant respectfully submits that neither *IBA* nor *KANAI* teaches or suggests the limitations highlighted above, and for this reason Claim 1 is nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*.

a. Overview of Claim 1

To aid in understanding Claim 1, consider the embodiment illustrated in Fig. 2 of the Application and described on pages 8 et seq., where a coordinator process for distributed transaction X 222 is responsible for coordinating other processes that are participating in distributed transaction X, such as slaves 232, 234, 236, and 252. Coordinator process 222 sends a publication request 224 to name service 202 to register the participants (e.g., slaves 232, 234, 236, and 252) for distributed transaction X. In response to publication request 224, name service 202 registers the participant data by storing replicated published data 219, 239, 259, 279 on database servers 210, 230, 250, 270, respectively.

As described in the Application, a “publication request is a request to make information available to a set of name service clients that request the information” and “the name service 202 supplies the data to any name service client requesting [the] data...” (Page 9, lines 5-12.) In order for the registered data to be available to the clients that request the data, the data can be registered at a number of times. For example, the data can be registered prior to the participants of distributed transaction X beginning their processing of distributed transaction X. As another example, the data can be registered as a result of the coordinator process for distributed transaction X 222 commencing the processing of distributed transaction X, but after individual participants beginning processing distributed transaction X. In general, the participant data can be registered at any time prior to a client making a request for some or all of the registered participant data. As one example, the request can be made to ascertain the participants in distributed transaction X to determine whether a deadlock has occurred in a distributed database transaction.

b. Summary of the Teachings of *IBA* and *KANAI*

*IBA*

*IBA* discloses a “system for detecting global deadlocks using wait-for graphs and identifiers of transactions related to the deadlocks in a distributed transaction processing system and a method of use therefore.” (Title.) Specifically, *IBA* discloses an approach for using a global deadlock detector as part of a transaction manager to detect deadlocks based on wait-for relations that arise from global transaction that stretch over a plurality of resource managers. (Abstract; Fig. 6 and Fig. 7.) By including a deadlock detector as part of the transaction manager that manages the global transactions, *IBA* solves the

problem of how to detect deadlocks stretching over resource managers between global transactions. (Col. 3, lines 25-30.) As described in the Background section of *IBA*, the conventional approach in this situation is to wait until a time-out occurs by monitoring execution time of one of the transactions whose processing has been substantially stopped due to a deadlock. (Col. 3, lines 31-35.)

The approach of *IBA* is to use the global deadlock detector with a wait-for graph (WFG) in the same manner as a local deadlock detector, namely to collect up wait-for requests from all of the global transactions, and then trace such requests to identify any loops that would indicate a deadlock. (See Col. 3, lines 43-50.) The major difference between the global deadlock detector and the local deadlock detector is that the former receives wait-for requests from all global transaction managed by the global transaction manager whereas the local deadlock detectors only receive wait-for requests for resources that are associated with the particular resource manager of which the local deadlock detector is a part. (See Col. 5, lines 11-24.)

Fig. 10A, Fig. 10B, Fig. 11 and the associated text of *IBA* illustrate the approach of using the global deadlock detector. "Fig. 10A shows the contents of node constituting a WFG and the contents comprise a chain portion and a SID portion as an identifier...Fig. 10B conceptually illustrates a tree structure of the chain portion." (Col. 10, lines 39-41 and 44-45.) Note that Fig. 10B lists the wait requests, such as "T2 is waiting for T1," etc. In Fig. 11, step S2 registers a request for registration of the wait-for relation of the global transaction. (Col. 11, lines 37-44.) If the wait-for relation is newly registered, then step S3 of Fig. 11 sets that wait-for relation as the starting point to trace the loop for deadlock determination. (Col. 11, lines 44-48.) Steps S4, S5, and S6 describe determining

whether a loop is detected, and if so, a transaction is selected to be cancelled in step S7, and if not, then the process returns to step S4. (Col. 11, lines 49-67.)

To summarize the approach of *IBA*, a global deadlock detector collects wait-for relations/requests from the global transactions. As each new wait request is received, a check is made to see if a loop can be detected. If so, a deadlock is identified and processing continues to clear the deadlock. If not, the process returns to wait for the next wait request. Note that in *IBA*: (A) only wait-for requests are stored by the global deadlock detector (e.g., Fig. 10B); (B) the wait-for requests are registered at the global deadlock detector as each wait-for condition is generated; and (C) there is no need to use time-outs since a check is made for a deadlock each time a wait-for request is registered at the global deadlock detector.

*KANAI*

*KANAI* discloses a “flexible distributed processing system capable of dealing with sophisticated conditions for selecting a server process.” (Abstract.) Generally speaking, *KANAI* is directed to an approach for selecting an appropriate server that is providing a desired service in response to a client inquiry. (Abstract) More specifically, *KANAI* teaches “a method of distributed processing among processors, where each processor having a server process for providing services and a service manager for managing the services provided by the server process...” in which the service manager registers each of the services provided by all the server processes along with an executability condition that is used by the service manager to select a server process that provides a service desired by a client such that the selected server process can satisfy the client’s request. (Col. 5, lines 12-29.)

c. Comparison of Claim 1 to *IBA* and *KANAI*

The applicant respectfully submits that neither *IBA* nor *KANAI* discloses the limitation in Claim 1 that recites “**said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction**”. In an Office Action mailed November 10, 2003, the Examiner asserted that this limitation was in fact disclosed in *IBA*. In response, in the Reply to Office Action mailed on February 9, 2004, the applicant provided a detailed argument as to why *IBA* does not disclose the above limitation. In the Final Office Action, with respect to the above limitation, the Examiner maintains the rejection of Claim 1 on the basis of *IBA*, but does not specifically point out where exactly in *IBA* was this limitation disclosed. Instead, in response to the argument the applicant made in the Reply mailed February 9, 2004, in the Final Office Action in the section “Response to Arguments”, paragraph 15, the Examiner states the following:

In response to the above-mentioned argument, In response to the above argument applicant’s interpretation of the prior art is noted, however, the features upon which applicant relies (i.e. “registration occurs without regard to whether a participant has made a wait-for-request, and the participant information is requested by a node, such as when such a node wishes to perform deadlock detection after expiration of a threshold period of time (e.g. such as time out condition being reached)”) are not recited in the rejected claim(s). This is not a suggestion of any sort. Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In Re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

While the applicant recognizes that features from the specification are not to be read into the claims, the applicant is at a loss to respond to the above paragraph. The Examiner has chosen to cite a miniscule portion of the applicant’s argument and take it out of context.



The applicant respectfully asserts the argument already made in the Reply to Office Action mailed February 9, 2004, and maintains the position that the limitation of Claim 1 requiring that **“said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction”**, is disclosed neither in *IBA* nor in *KANAI*.

Specifically, as asserted in the Reply mailed February 9, 2004, Claim 1 requires that **“said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction...”** Thus, participant data is registered in the name service when the participants commence participating in the distributed transaction, so that later such information can be used by another node, such as in detecting a deadlock. For example, when trying to detect deadlocks, upon expiration of a threshold period of time (such as a time-out), a deadlock handler can determine which participants are involved in the distributed transaction by requesting the participant data from the name service, and then constructing a wait-for graph, thereby avoiding the need to use a broadcast query to all nodes to determine which nodes are participating in the distributed transaction and thus may be involved in the deadlock. (Application, page 12, lines 7-24.)

Note that in the approach of Claim 1: (A) participant data is registered with the name service in response to the participants commencing participation in the distributed transaction, (B) registration occurs without regard to whether a participant has made a wait-for request; and (C) the participant information is requested by a node, such as when such a node wishes to perform deadlock detection after expiration of a threshold period of time (e.g., such as a time out condition being reached).

Thus, the approach of Claim 1 is fundamentally different from that of *IBA* because in Claim 1, participant data is registered in the name service when the participants commence participation in the distributed transaction, whereas in *IBA*, the approach is to register wait-for relations as the wait-for requests are generated. This means that with the approach of Claim 1, participant data is registered that does not include any wait-for requests, although such wait requests for those participants can be generated later. In contrast, in *IBA*, participant data for participants that have not generated a wait request is not registered.

Therefore, the applicant respectfully submits that *IBA*, either alone or in combination with *KANAI*, fails to disclose, teach, suggest, or in any way render obvious registering participant data where “**said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction...**” as featured in Claim 1.

d. Conclusion about the patentability of Claim 1 under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*

For the above reasons, the applicant respectfully submits that Claim 1 is patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* and *KANAI* because neither *IBA* nor *KANAI*, separately or in combination, teaches or suggests all of the limitations recited in Claim 1.

2. **Dependent Claims 2, and 8-9 are patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* because they depend from Claim 1**

Dependent Claims 2, 8, and 9 depend from independent Claim 1. Claims 2, 8, and 9 are therefore allowable for the reasons given above with respect to Claims 1.

3. **Dependent Claim 3 is patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* because it depends from Claim 1 and because it includes additional limitations not described in *IBA* or *KANAI***

Dependent Claim 3 depends from independent Claim 1, and is therefore allowable for the reasons given above with respect to Claim 1. Further, Claim 3 recites an additional limitation which requires “**registering in said name service participant data that identifies which database servers of a plurality of database servers are participating in said distributed transaction.**”

The Final Office Action alleges that the above limitation requiring registering of database servers participating in a distributed transaction is disclosed in *KANAI*, in col. 1, lines 51-55. In *KANAI*, col. 1, lines 51-55 recite the following:

In the transaction processing, the processings are executed in units of transactions, and in each transaction, a series of processings are completed by executing a plurality of processes related to application programs, database management systems, etc.

While the applicant admits that this paragraph mentions that processes may execute in database management systems, the applicant respectfully submits that no where in this paragraph is there any mentioning that database servers participating in a **distributed transaction** may be registered with a name service, as is required by the express limitation recited in Claim 3. In fact, nowhere else in *KANAI* is there a description of a process step where a database server participating in a distributed transaction is registering with a name service.

For this reason, the applicant respectfully submits that the limitation in Claim 3 which requires “**registering in said name service participant data that identifies which database servers of a plurality of database servers are participating in said distributed transaction**” is not disclosed in *IBA* or *KANAI*, and for this reason Claim 3 is patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*.

4. **Dependent Claims 4 and 5 are patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* because they depend from Claim 1 and because they include additional limitations not described in *IBA* or *KANAI***

Dependent Claims 4 and 5 depend from independent Claim 1, and are therefore allowable for the reasons given above with respect to Claim 1. Further, Claim 4 recites an additional step that requires “**causing updates to said participant data to identify a new participant in said distributed transaction.**”

The Final Office Action alleges that *IBA* discloses this limitation in col. 9, lines 25-45. However, a closer look at this passage in *IBA* reveals that what the passage discloses is the process by which a Local Lock Manager (LLM) receives a request to acquire or release hardware resources to lock or unlock resources from an application program. (*IBA*, col. 9, lines 24-26). Significantly, if the resource cannot be locked, the identifier of the transaction waiting for locking, and the identifier of the transaction currently occupying the resource are notified to the Global Deadlock Detector (GDD). (*IBA*, col. 9, lines 29-33). The GDD then registers the information in the WFG graph and

determines whether or not there are deadlocks between global transactions. (*IBA*, col. 9, lines 40-45).

Thus, in contrast to the express limitation recited in Claim 4, the above cited passage does NOT disclose that any information identifying a new participant in a **distributed transaction** is passed to the GDD for registration in the WFG graph table (the name service). In fact, there is no mention at all of a distributed transaction, and all the passage describes is how separate and distinct global transactions can acquire or release locks on resources. There is not even a hint in *IBA* that these global transactions may be a part of one specific distributed transaction, as is expressly required by the language recited in Claim 4.

For this reason, the applicant respectfully submits the limitation in Claim 4 which requires “**causing updates to said participant data to identify a new participant in said distributed transaction**” is not disclosed in *IBA* or *KANAI*, and for this reason Claim 4 is patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*. Furthermore, dependent Claim 5 depends from Claim 4, and incorporates all of the limitations recited in Claim 4. For this reason, Claim 5 is also patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*.

5. **Dependent Claims 6 and 7 are patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* because they depend from Claim 1 and because they include additional limitations not described in *IBA* or *KANAI***

Dependent Claims 6 and 7 depend from independent Claim 1, and are therefore allowable for the reasons given above with respect to Claim 1. Further, both Claims 6 and 7 expressly recite an additional limitation that requires that “**said distributed transaction is a distributed database transaction.**”

The Examiner alleges in the Final Office Action that this limitation is disclosed in *IBA* (col. 9, lines 24-39 with respect to the distributed transaction), and *KANAI* (col. 1, lines 51-56 with respect to the distributed transaction being a database transaction). However, as shown above in the discussion about Claim 4, the cited passage from *IBA* (namely, col. 9, lines 24-39) does not disclose that registering of information in the WFG table (the name service) involves any data about any participants of any one particular distributed transaction. Moreover, as shown above in the discussion about Claim 3, the cited passage from *KANAI* (namely col.1, lines 51-56) does not disclose that the transaction in question is a database transaction distributed among more than one database servers.

In contrast, Claims 6 and 7 both require specifically that the distributed transaction be a distributed database transaction. For these reasons, since neither *IBA* nor *KANAI* discloses the limitation recited in Claims 6 and 7 requiring that the distributed transaction be a “**distributed database transaction**”, both Claims 6 and 7 are patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*.

In addition, Claim 6 requires that the step of registering include “**registering participant data that identifies which database servers of a plurality of database servers are participating in said distributed database transaction**”. In other words, Claim 6 requires that the participants be database servers. In the Final Office Action, the

Examiner asserts the same passage from *KANAI* (col. 1, line 51-56) cited to support the rejection of Claim 3. However, as shown above in the discussion about Claim 3, the Applicant does not believe that the cited passage supports the disclosure of database servers as participants in a distributed database transaction, which is required by the language recited in Claim 6.

For this additional reason, since neither *IBA* nor *KANAI* discloses the limitation recited in Claim 6 requiring “**registering participant data that identifies which database servers of a plurality of database servers are participating in said distributed database transaction**”, Claim 6 is patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*.

6. **Independent Claim 11 is patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* for the same reasons discussed with respect to Claim 1**

Claim 11 currently reads as follows:

A computer readable medium carrying one or more sequences of one or more instructions for determining participants of a distributed transaction in a distributed system, the one or more sequences of one or more instructions including instructions which, when executed by one or more processors, cause the one or more processors to perform the steps of:  
registering, in a name service, participant data that identifies a plurality of  
participants that are participating in said distributed transaction,  
wherein **said step of registering occurs in response to said**

**plurality of participants commencing participation in said  
distributed transaction;**

causing a node that requires information about participants in said  
distributed transaction to request said participant data from said  
name service.” (Emphasis added).

As is evident, Claim 11 includes the same limitations as recited in Claim 1. Thus, for the same reasons discussed above with respect to Claim 1, the applicant respectfully submits that Claim 11 is patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI* because neither *IBA* nor *KANAI*, either separately or in combination, teaches or suggests all of the limitations recited in Claim 11.

7. **Dependent Claims 12, and 18-19 are patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* because they depend from Claim 11**

Dependent Claims 12, 18, and 19 depend from independent Claim 11. Claims 12, 18, and 19 are therefore allowable for the reasons given above with respect to Claims 11.

8. **Dependent Claim 13 is patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* because it depends from Claim 11 and because it includes additional limitations not described in *IBA* or *KANAI***

Dependent Claim 13 depends from independent Claim 11, and is therefore allowable for the reasons given above with respect to Claims 11. Further, Claim 13



recites an additional limitation which requires “**registering in said name service participant data that identifies which database servers of a plurality of database servers are participating in said distributed transaction.**” This limitation is the same limitation discussed with respect to Claim 3, and for the reasons given in the discussion about Claim 3, Claim 13 is allowable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*.

9. **Dependent Claims 14 and 15 are patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* because they depend from Claim 11 and because they include additional limitations not described in *IBA* or *KANAI***

Dependent Claims 14 and 15 depend from independent Claim 11, and are therefore allowable for the reasons given above with respect to Claim 11. Further, Claims 14 and 15 recite additional limitations that were recited in, and discussed with respect to, Claims 4 and 5. Therefore, for the reasons given with respect to Claims 4 and 5, the applicant respectfully submits that Claims 14 and 15 are patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*.

10. **Dependent Claims 16 and 17 are patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* because they depend from Claim 11 and because they include additional limitations not described in *IBA* or *KANAI***

Dependent Claims 16 and 17 depend from independent Claim 11, and are therefore allowable for the reasons given above with respect to Claim 11. Further, Claims 16 and 17 recite additional limitations that were recited in, and discussed with respect to, Claims 6 and 7. Therefore, for the reasons given with respect to Claims 6 and 7, the applicant respectfully submits that Claims 16 and 17 are patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*.

11. **Independent Claim 22 is patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* for the same reasons discussed with respect to Claim 1**

In the Final Office Action, the Examiner rejected Claim 22 on the ground that Claim 22 comprised limitations discussed with respect to Claims 1-9. (See Final Office Action, page 4). Claim 22 currently reads as follows:

A method for determining a plurality of participants that are participating in a distributed transaction, the method comprising the computer-implemented steps of:

in response to said plurality of participants commencing participation in said distributed transaction, receiving **first data** that identifies said plurality of participants;

in response to receiving said first data, registering said **first data**;

receiving request from a node;

in response to said request from said node, providing **second data** to said node,

wherein said **second data** includes at least part of said **first data**.

While Claim 22 comprises some of the same limitations recited in Claim 1, the applicant respectfully points out that Claim 22 includes the additional limitations “**first data that identifies said plurality of participants**”, and “**providing second data to said node, wherein said second data includes at least part of said first data.**” Since the rejection does not specify where exactly in *IBA* and *KANAI* are these two limitations to be found, and since the applicant cannot find where in *IBA* or *KANAI* are these limitations disclosed either expressly or inherently, the applicant respectfully submits that Claim 22 is nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*. For this reason, the applicant respectfully submits that Claim 22 is allowable.

Further, with respect to the limitations common to Claims 1 and 22, and for the reasons discussed above with respect to Claim 1, the applicant respectfully submits that Claim 22 is patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*.

Moreover, since Claims 23-26 depend from Claim 22, the applicant submits that Claims 23-26 are patentable for the same reasons Claim 22 is patentable.

**12. Independent Claim 27 is patentable under 35 U.S.C. §103(a) as being nonobvious over *IBA* in view of *KANAI* for the same reasons discussed with respect to Claim 22**

Claim 27 is the computer-readable version of Claim 22, and contains limitations similar to the limitations discussed with respect to Claims 1 and 22. Therefore, for the same reasons discussed above with respect to Claims 1 and 22, the applicant respectfully

submits that Claim 27 is patentable as nonobvious under 35 U.S.C. §103(a) over *IBA* in view of *KANAI*.

Further, since Claims 28-31 depend from Claim 27, the applicant submits that Claims 28-31 are patentable for the same reasons Claim 27 is patentable.

**C. Claims 1 and 11 are patentable under 35 U.S.C. §103(a) as being nonobvious over *VAN DEN BERG* in view of *NECHES* because the combination of *VAN DEN BERG* and *NECHES* does not disclose, teach or suggest all of the claimed limitations**

In the Final Office Action, the Examiner maintains the rejection of Claims 1 and 11 that was raised in an earlier Office Action. The applicant respectfully submits that neither *VAN DEN BERG* nor *NECHES* discloses the limitation recited in both Claims 1 and 11 which requires that “... **said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction ...**.”

*VAN DEN BERG*

*VAN DEN BERG* discloses a “distributed processing system that includes a distributed resource manager which detects dependencies between transactions caused by conflicting lock requests.” (Abstract.) Generally speaking, *VAN DEN BERG* is directed to an approach for using wait-for graphs to detect deadlocks in which failure resilience “is achieved by duplicating between agents and servers, rather than by duplicating the servers. As a result, the number of messages between agents and servers in normal operation is not increased.” (Abstract.) More specifically, *VAN DEN BERG* teaches a

deadlock detection approach that is similar to *IBA*, namely that dependencies between transactions are “maintained in a queue of lock requests that cannot be immediately granted, and for detecting dependencies between transactions caused by conflicting lock requests (e.g., using a wait-for graph to track wait-for relations and thereby detect deadlocks). (Fig. 3; Col. 3, lines 1-15; Col. 18, lines 4-7.) Again as in *IBA*, the approach in *Van Den Berg* is to only track the lock requests that cannot be granted, not to register participant data in response to the participants commencing participation in the distributed transaction, as featured in Claims 1 and 11. As a result, in the approach of *VAN DEN BERG*, the information being tracked does not include participants that are not waiting on a resource, as is the case in the approach of Claims 1 and 11.

Thus, the Applicant respectfully submits that *VAN DEN BERG*, either alone or in combination with the other cited references, fails to disclose, teach, suggest, or in any way render obvious registering participant data where **“said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction...”** as featured in Claims 1 and 11.

*NECHES*

*NECHES* discloses a “system using a sorting network to intercouple multiple processors so as to distribute priority messages to all processors...” (Abstract.) Generally speaking, *NECHES* is directed to an approach for achieving greater flexibility as to intercommunications and control by using transaction numbers to identify tasks and track the status of the tasks using prioritized responses and controlling the flow of messages. (Abstract.) More specifically, *NECHES* teaches an approach in which the “many message routing, mode control and status indication functions required for a complex and

versatile multiprocessor system are provided in accordance with the invention by a unique combination of message organization and traffic controlling interface circuits functioning with the active logic network.” (Col. 3, lines 20-25.) *NECHES* uses “transaction identities” and “a single query to all processors” to obtain “the global status of the system.” (Col. 3, lines 41-54.) Thus, *NECHES* is similar to *KANAI* in that both are generally directed to managing the distribution of tasks among multiple processors and tracking the status of such tasks.

However, there is nothing in *NECHES* that discloses, teaches, suggests, or renders obvious registering participant data where “**said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction...**,” as featured in Claims 1 and 11. Furthermore, *NECHES* is only concerned with message routing, mode control, and status indication functions for a multiprocessor system, and therefore, *NECHES* is not concerned with a “**distributed transaction**” as featured in Claims 1 and 11.

Thus, the Applicant respectfully submits that *NECHES*, either alone or in combination with *VAN DEN BERG*, fails to disclose, teach, suggest, or in any way render obvious registering participant data where “**said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction...**,” as featured in Claims 1 and 11.

Conclusion with regards to the patentability of Claims 1 and 11 in light of *VAN DEN BERG* and *NECHES*

For the foregoing reasons, the applicant respectfully asserts that Claims 1 and 11 are patentable as nonobvious under 35 U.S.C. §103(a) over *VAN DEN BERG* in view of

*NECHES* because neither *VAN DEN BERG* nor *NECHES*, taken alone or in combination, teaches, describes, or suggests all of the limitations recited in Claims 1 and 11.

**D. Claims 1, 11, 22, and 27, and all claims that depend from them, are patentable under 35 U.S.C. §103(a) because the Office Actions lack the requisite suggestion, teaching, or motivation to combine *IBA* and *KANAI* on one hand, or *VAN DEN BERG* and *NECHES* on the other**

The Final Office Action states that it would have been obvious to one with ordinary skill in the art at the time the invention was made to combine *IBA* and *KANAI* in the first obviousness rejection of Claims 1, 11, 22, and 27 and to combine *VAN DEN BERG* and *NECHES* in the second obviousness rejection of Claims 1 and 11. However, notwithstanding the fact that none of *IBA*, *KANAI*, *VAN DEN BERG* or *NECHES* disclose registering participant identifying data where “**said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction...**,” as featured in Claims 1 and 11 (and as substantially recited in Claims 22 and 27), the applicant respectfully submits that there is nothing in any of *IBA*, *KANAI*, *VAN DEN BERG* or *NECHES* that teaches, suggests, or motivates combining their respective teachings.

As stated in the Federal Circuit decision *In re Dembiczak*, 175 F.3d 994, 50 USPQ.2d 1617 (Fed. Cir. 1999), (citing *Gore v. Garlock*, 220 USPQ 303, 313 (Fed. Cir. 1983)), “it is very easy to fall victim to the insidious effect of the hindsight syndrome where that which only the inventor taught is used against its teacher.” *Id.* The Federal Circuit stated in *Dembiczak* “that the best defense against subtle but powerful attraction

of a hindsight-based obviousness analysis is rigorous application of the requirement for a showing of the teaching or suggestion to combine prior art references.” *Id.* Thus, the Federal Circuit explains that a proper obviousness analysis requires “***particular factual findings*** regarding the locus of the suggestion, teaching, or motivation to combine prior art references.” *Id.* (emphasis added).

In particular, the Federal Circuit states:

“We have noted that evidence of a suggestion, teaching, or motivation to combine may flow from the prior art references themselves, the knowledge of one of ordinary skill in the art, or, in some cases, from the nature of the problem to be solved...although ‘the suggestion more often comes from the teachings of the pertinent references’...The range of sources available, however, does ***not diminish the requirement for actual evidence***. That is, the ***showing must be clear and particular***...Broad conclusory statements regarding the teaching of multiple references, standing alone, are not ‘evidence.’” *Id.* (emphasis added; internal citations omitted).

None of *IBA*, *KANAI*, *VAN DEN BERG* or *NECHES* shows any suggestion, teaching, or motivation to combine their teachings, nor does the Final Office Action provide a “clear and particular” showing of the suggestion, teaching, or motivation to combine their teachings. In fact, the only motivation provided in the Final Office Action, as well as in the prior Office Actions, are the broad conclusory statements that by combining features of those references, one may achieve the benefits achieved from the invention as described and claimed in the application. It is respectfully submitted that such a hindsight observation is not consistent with the Federal Circuit’s requirement for “particular factual findings,” and thus that both the obviousness rejection of Claims 1, 11, 22 and 27 on the basis of *IBA* in view of *KANAI* and the obviousness rejection of Claims 1 and 11 on the basis of *VAN DEN BERG* in view of *NECHES* are improper.



**E. Claim 1 is patentable under 35 U.S.C. §102(b) because it is not anticipated by *NECHES***

“A claim is anticipated only if each and element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference.” *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987); *see also* MPEP §2131.

In the Final Office Action, the Examiner rejected Claim 1 as being anticipated under 35 U.S.C. §102(b) by *NECHES*. The rejection reads as follows (quoting verbatim):

Regarding claim 1, Neches teaches a method comprising the steps of:

registering in storage means (“name service”) participant data  
identifies a plurality of participants that are participating a distributed  
operation, wherein the step of registering occurs in response to a request to  
a plurality of process(es) to determine which are participating in said  
distributed operation (see col 3/lines 41-66, col 4/lines 13-38, col 13/lines  
59-63, col 14/lines 13-34 and col 15/lines 57-col 16/line 19).

As is evident from this citation, the Final Office Action fails point out anything in *NECHES* that corresponds to the step of **“causing a node that requires information about participants in said distributed transaction to request said participant data from said name service.”** Furthermore, the applicant fails to see anything in *NECHES* that inherently describes this limitation. For this reason, the applicant respectfully submits that *NECHES* does not describe, either expressly or inherently, the above limitation.

Moreover, as pointed out in the discussion above regarding the 35 U.S.C. §103(a) rejection of Claims 1 and 11 on the basis of *VAN DEN BERG* and *NECHES*, *NECHES* fails to describe the particular limitation recited in Claim 1 requiring “**said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction...**”.

For these reasons, the applicant respectfully submits that *NECHES* does not describe all of the limitations recited in Claim 1, and hence Claim 1 is patentable as not being anticipated under 35 U.S.C. §102(b) by *NECHES*.

**F. Claims 1 and 11 are patentable under 35 U.S.C. §101 over a statutory double patenting rejection on the basis of Claims 1-3 in *FISCHER* (U.S. Patent No. 6, 594,702) because the present application and *FISCHER* do not claim the same invention**

In determining whether a statutory basis for a double patenting rejection exists, the question to be asked is: Is the same invention being claimed twice? 35 U.S.C. §101 prevents two patents from issuing on the same invention. “Same invention” means identical subject matter. *Miller v. Eagle Mfg. Co.*, 151 U.S. 186 (1984); *In Re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In Re Ockert*, 245 F.2d 467, 114 USPQ 330 (CCPA 1957). A reliable test for double patenting under 35 U.S.C. §101 is whether a claim in the application could be literally infringed without literally infringing a corresponding claim in the patent. *In Re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); MPEP §804. Is there an embodiment of the invention that falls within the scope of one claim but not the other? If there is such an embodiment, then identical subject

matter is not defined by both claims and statutory double patenting would not exist.  
MPEP §804.

The invention claimed in *FISCHER* relates to managing the size and accessibility of a name service. More specifically, *FISCHER* claims methods for managing a name service that includes published data that is associated with one or more duration entities and one or more modification entities. (*FISCHER*, col. 2, lines 44-47).

A modification entity is an entity permitted by a name service to alter published data. A duration entity is an entity whose duration is used to dictate the duration of published data. (*FISCHER*, col. 5, lines 59-62). By associating name entries with a duration entity and a modification entity, the name service is able to more efficiently manage access to published data and the amount of memory needed to store it. (*FISCHER*, col. 5, lines 62-65). The *FISCHER* invention is claimed in Claims 1-3 as follows:

1. A method of exchanging information between clients on a computer system, the method comprising the steps of:

a name service receiving from a client:

published data,

key information that identifies one or more keys associated with said  
published data, and

**duration entity information that identifies one or more duration  
entities associated with published data;**

**wherein said one or more duration entities are each an entity that has a finite  
duration and that terminates after a the finite duration;**

said name service storing an entry that associates said published data with said  
one or more keys;

**for a period of time, said name service supplying said published data to clients  
requesting data associated with said one or more keys;**

**said name service ceasing to supply said published data to clients after said period of time; and**

**wherein said period of time is based on at least one duration of said one or more duration entities.**

2. The method of claim 1, wherein:

the step of receiving from a client duration entity information includes receiving information that identifies a process associated with said published data; and

**said period of time expires when said process expires.**

3. The method of claim 1, wherein:

the step of receiving from a client duration entity information includes receiving information that identifies a transaction; and

**wherein said period of time expires when said transaction ends.**

Thus, in order for a statutory double-patenting rejection to be sustained, Claims 1 and 11 of the present invention must be the “same invention” claimed in claims 1-3 in *FISCHER*. A closer look at Claim 1 reveals that there is an **embodiment** that literally falls within the scope of Claim 1 but which does NOT literally fall within the scope of claims 1-3 of the *FISCHER* patent. Specifically, consider the following hypothetical method:

A method of determining participants of a distributed transaction in a distributed system, the method comprising the steps of:

registering, in a name service, participant data that identifies a plurality of participants that are participating in said distributed transaction,

wherein said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction, and

**wherein said plurality of participants do NOT provide any durational information that indicates the participant has a finite duration;**

causing a node that requires information about participants in said distributed transaction to request said participant data from said service; and  
**said name service continuing to provide said participant data until said plurality of participants deregisters their corresponding participant data with said name service.**

A comparison of this hypothetical method to Claim 1 in the present application reveals that indeed the hypothetical method is within the scope of Claim 1 because it literally incorporates ALL of the limitations recited in Claim 1 (the non-highlighted portion). The highlighted limitations do not take the hypothetical method outside the scope of Claim 1 because Claim 1 is open-ended and covers any embodiment that includes all of the claim limitations.

On the other hand, comparing the hypothetical method to Claims 1-3 of the *FISCHER* patent reveals that the hypothetical method does NOT literally infringe any of these three claims. Specifically, (1) the hypothetical method requires that participants in a distributed transaction **“do NOT provide any durational information that indicates the participant has a finite duration”** to the name service as required by claim 1 of the *FISCHER* patent; and (2) the hypothetical method requires the name service to continue providing participant data **“until said plurality of participants deregisters their corresponding participant data with said name service”**, while claim 1 of the *FISCHER* patent expressly requires that **“for a period of time, said name service supplying said published data to clients ..., said name service ceasing to supply said published data to clients after said period of time; and wherein said period of time is based on at least one duration of said one or more durational entities”**. In other words, the hypothetical method does not include any “durational” participants, and the

name service provides participant data until the participant deregisters, while claim 1 of the *FISCHER* patent requires that participants provide durational information, and deregistering is done by the name service based on this durational information.

Moreover, the hypothetical method requires **the participant to deregister** its published data from the name service, while claims 2 and 3 of the *FISCHER* patent require that the **name service deregister** the participant data when the participant (a process or transaction) expires.

Thus, applicant respectfully submits that, under the *In Re Vogel* test, there exists an embodiment literally within the scope of Claim 1 of the present invention that does not literally infringe claims 1-3 of the *FISCHER* patent. Furthermore, substantially the same hypothetical embodiment can be constructed under Claim 11 of the present invention because Claim 11 involves the same limitations as Claim 1. Therefore, applicant respectfully submits that Claims 1 and 11 are patentable over the statutory double-patenting rejection under 35 U.S.C. §101 on the basis of the *FISCHER* patent.

**F. Claim 1 is patentable over the obviousness-type double patenting rejection on the basis of Claims 1-3 in *FISCHER* (U.S. Patent No. 6, 594,702) in view of *KANAI* because *KANAI* does not teach at least one limitation recited in Claim 1 that is not claimed in *FISCHER***

A double patenting rejection of the obviousness-type is “analogous to [a failure to meet] the nonobviousness requirement of 35 U.S.C. 103” except that the patent principally underlying the double-patenting rejection is not considered prior art. *In Re Braithwaite*, 379 F.2d 594, 154 USPQ 29 (CCPA 1967); MPEP §804. Therefore, any

analysis employed in an obviousness-type double patenting rejection parallels the guidelines for analysis of a 35 U.S.C. §103 obviousness determination, (*In Re Braat*, 937 F.2d 589, 19 USPQ2d 1289 (Fed. Cir. 1991); *In Re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); MPEP §804), except that “[w]hen considering whether the invention defined in a claim of an application is an obvious variation of the invention defined in the claim of a patent, the disclosure of the patent may not be used as prior art.” (MPEP §804, section II, subsection B-1).

In the Final Office Action, the Examiner rejected Claim 1 for obviousness-type double patenting over claims 1-3 of the *FISCHER* patent in view of *KANAI*.

The applicant respectfully submits that subject matter that is disclosed but not claimed in *FISCHER* is unavailable as prior art on the basis of *In Re Braithwaite*. Thus, in order for the obviousness-type double patenting rejection to be sustained, *KANAI* must teach all of the limitations recited in Claim 1 that are not recited in claims 1-3 of *FISCHER*.

However, claims 1-3 of *FISCHER* (cited above) do not include at least one limitation of Claim 1 – the limitation requiring that “**said step of registering occurs in response to said plurality of participants commencing participation in said distributed transaction.**” Furthermore, as shown in the discussion above with respect to the obviousness rejection of Claim 1 on the basis of *IBA* in view of *KANAI*, *KANAI* does not teach this same limitation. For this reason, the applicant respectfully submits that Claim 1 is patentable over the obviousness-type double patenting rejection on the basis claims 1-3 of *FISCHER* in view *KANAI*.

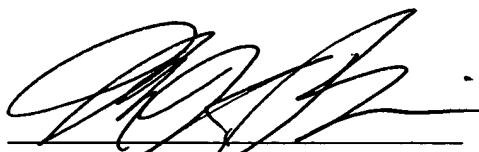
In sum, the appellant respectfully submits that the imposed rejections under 35 U.S.C. §112, first paragraph, the rejections under 35 U.S.C. §103(a) on the basis of *IBA* in view of *KANAI*, the rejections under 35 U.S.C. §103(a) on the basis of *VAN DEN BERG* in view of *NECHES*, the rejection under 35 U.S.C. §102(b) on the basis of *NECHES*, the statutory double patenting rejection under 35 U.S.C. §101 on the basis of *FISCHER*, and the obviousness-type double patenting rejection over claims 1-3 of *FISCHER* in view of *KANAI* are all **improper** and respectfully solicits the Honorable Board to **reverse** each of these rejections.

To the extent necessary to make this Appeal Brief timely filed, the Applicant petitions for an extension of time under 37 C.F.R. § 1.136.

If any applicable fee is missing or insufficient, throughout the pendency of this application, the Commissioner is hereby authorized to any applicable fees and to credit any overpayments to our Deposit Account No. 50-1302.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP



Marcel K. Bingham  
Reg. No. 42,327

1600 Willow Street  
San Jose, CA 95125  
(408) 414-1080, ext. 206

**Date:** September 27, 2004

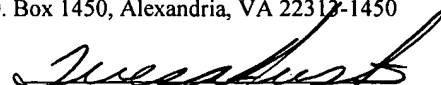
Facsimile: (408) 414-1076

**CERTIFICATE OF MAILING**

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Mail Stop Appeal Brief – Patents, Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22312-1450

on September 27, 2004

by





## CLAIMS APPENDIX

1 (Previously Presented) A method of determining participants of a distributed transaction  
2 in a distributed system, the method comprising the steps of:  
3 registering, in a name service, participant data that identifies a plurality of participants  
4 that are participating in said distributed transaction, wherein said step of  
5 registering occurs in response to said plurality of participants commencing  
6 participation in said distributed transaction; and  
7 causing a node that requires information about participants in said distributed  
8 transaction to request said participant data from said name service.

1 2. (Previously Presented) The method of Claim 1, wherein the step of causing a node  
2 includes causing said node to retrieve said participant data in response to said node  
3 performing deadlock detection.

1 3. (Previously Presented) The method of Claim 1, wherein:  
2 the step of registering includes registering in said name service participant data that  
3 identifies which database servers of a plurality of database servers are  
4 participating in said distributed transaction.

1 4. (Previously Presented) The method of Claim 1, further including the step of causing  
2 updates to said participant data to identify a new participant in said distributed  
3 transaction.

1 5. (Previously Presented) The method of Claim 4, wherein:

2 said distributed transaction is a distributed database transaction being executed by a  
3 set of processes coordinated by a coordinator process;  
4 the method further includes the step of said coordinator process causing a new process  
5 on a database server to participate in said distributed database transaction; and  
6 the step of causing updates to said participant data includes said coordinator process  
7 causing updates to said participant data in response to said new process  
8 participating in said distributed database transaction.

1 6. (Previously Presented) The method of Claim 1, wherein:

2 said distributed transaction is a distributed database transaction;  
3 the step of registering includes registering participant data that identifies which  
4 database servers of a plurality of database servers are participating in said  
5 distributed database transaction; and  
6 the step of causing a node includes causing a node that requires information about  
7 participants in said distributed database transaction to retrieve said participant  
8 data from said name service.

1 7. (Previously Presented) The method of Claim 1, wherein:

2 said distributed transaction is a distributed database transaction;  
3 the method further includes the step of assigning a transaction identifier to said  
4 distributed database transaction;  
5 the step of registering includes registering, in said name service, data that associates  
6 said participant data with said transaction identifier; and

7 the step of causing a node includes causing a node to request, from said name service,  
8 published data associated with said transaction identifier.

1 8. (Previously Presented) The method of Claim 1, wherein the steps further include said  
2 name service receiving a request from a first process to supply said participant data,  
3 wherein said name service and said first process reside on said node.

1 9. (Previously Presented) The method of Claim 8, wherein the step of causing a node  
2 includes said name service retrieving said participant data from one or more data  
3 structures residing on said node in response to receiving said request.

1 10. (Canceled)

1 11. (Previously Presented) A computer-readable medium carrying one or more sequences of  
2 one or more instructions for determining participants of a distributed transaction in a  
3 distributed system, the one or more sequences of one or more instructions including  
4 instructions which, when executed by one or more processors, cause the one or more  
5 processors to perform the steps of:

6 registering in a name service participant data that identifies a plurality of participants  
7 that are participating in said distributed transaction, wherein said step of  
8 registering occurs in response to said plurality of participants commencing  
9 participation in said distributed transaction; and

10 causing a node that requires information about participants in said distributed  
11 transaction to request said participant data from said name service.

1 12. (Previously Presented) The computer-readable medium of Claim 11, wherein the step of  
2 causing a node includes causing said node to retrieve said participant data in response to  
3 said\_node performing deadlock detection.

1 13. (Previously Presented) The computer-readable medium of Claim 11, wherein:  
2 the step of registering includes registering in said name service participant data that  
3 identifies which database servers of a plurality of database servers are  
4 participating in said distributed transaction.

1 14. (Previously Presented) The computer-readable medium of Claim 11, further including the  
2 step of causing updates to said participant data to identify a new participant in said  
3 distributed transaction.

1 15. (Previously Presented) The computer-readable medium of Claim 14, wherein:  
2 said distributed transaction is a distributed database transaction being executed by a  
3 set of processes coordinated by a coordinator process;  
4 the computer-readable medium further includes sequences of instructions for  
5 performing the step of said coordinator process causing a new process on a  
6 database server to participate in said distributed database transaction; and  
7 the step of causing updates to said participant data includes said coordinator process  
8 causing updates to said participant data in response to said new process  
9 participating in said distributed database transaction.

1 16. (Previously Presented) The computer-readable medium of Claim 11, wherein:  
2 said distributed transaction is a distributed database transaction;

3 the step of registering includes registering participant data that identifies which  
4 database servers of a plurality of database servers are participating in said  
5 distributed database transaction; and  
6 the step of causing a node includes causing a node that requires information about  
7 participants in said distributed database transaction to retrieve said participant  
8 data from said name service.

1 17. (Previously Presented) The computer-readable medium of Claim 11, wherein:  
2 said distributed transaction is a distributed database transaction;  
3 the steps further include the step of assigning a transaction identifier to said  
4 distributed database transaction;  
5 the step of registering includes registering in said name service data that associates  
6 said participant data with said transaction identifier; and  
7 the step of causing a node includes causing a node to request, from said name service,  
8 published data associated with said transaction identifier.

1 18. (Previously Presented) The computer-readable medium of Claim 11, wherein the steps  
2 further include said name service receiving a request from a first process to supply said  
3 participant data, wherein said name service and said first process reside on said node.

1 19. (Previously Presented) The computer-readable medium of Claim 18, wherein the step of  
2 causing a node includes said name service retrieving said participant data from one or  
3 more data structures residing on said node in response to receiving said request.

1 20. (Canceled)

1 21. (Canceled)

1 22. (Previously Presented) A method for determining a plurality of participants that are  
2 participating in a distributed transaction, the method comprising the computer-  
3 implemented steps of:

4 in response to said plurality of participants commencing participation in said  
5 distributed transaction, receiving first data that identifies said plurality of  
6 participants;

7 in response to receiving said first data, registering said first data;  
8 receiving a request from a node;

9 in response to said request from said node, providing second data to said node,  
10 wherein said second data includes at least part of said first data.

1 23. (Previously Presented) The method of Claim 22, wherein a name service performs the  
2 steps of receiving said first data, registering said first data, receiving said request, and  
3 providing said second data.

1 24. (Currently Amended) The method of Claim 22, wherein said node uses said ~~information~~  
2 second data to determine whether a deadlock exists, and wherein said request is received  
3 after a particular participant of said plurality of participants has waited for a threshold  
4 period of time.

1 25. (Previously Presented) The method of Claim 22, wherein:

2 said distributed transaction is a distributed database transaction; and

3 said first data identifies one or more database servers of a plurality of database servers

4 that are participating in said distributed database transaction.

1 26. (Previously Presented) The method of Claim 22, wherein:

2 said plurality of participants includes all participants in the distributed transaction; and

3 said first data identifies said all participants in the distributed transaction.

1 27. (Previously Presented) A computer-readable medium carrying one or more sequences of

2 one or more instructions for determining a plurality of participants that are participating

3 in a distributed transaction, the one or more sequences of one or more instructions

4 including instructions which, when executed by one or more processors, cause the one or

5 more processors to perform the steps of:

6 prior to said plurality of participants commencing participation in said distributed

7 transaction, receiving first data that identifies said plurality of participants;

8 in response to receiving said first data, registering said first data;

9 receiving a request from a node;

10 in response to said request from said node, providing second data to said node,

11 wherein said second data includes at least part of said first data.

1 28. (Previously Presented) The computer-readable medium of Claim 27, wherein a name

2 service performs the steps of receiving said first data, registering said first data, receiving

3 said request, and providing said second data.

1   29.   (Currently Amended) The computer-readable medium of Claim 27, wherein said  
2           node uses said ~~information~~ second data to determine whether a deadlock exists,  
3           and wherein said request is received after a particular participant of said plurality  
4           of participants has waited for a threshold period of time.

1   30.   (Previously Presented) The computer-readable medium of Claim 27, wherein:  
2           said distributed transaction is a distributed database transaction; and  
3           said first data identifies one or more database servers of a plurality of database  
4           servers that are participating in said distributed database transaction.

1   31.   (Previously Presented) The computer-readable medium of Claim 27, wherein:  
2           said plurality of participants includes all participants in the distributed  
3           transaction; and  
4           said first data identifies said all participants in the distributed transaction.